What is claimed is:

1. A method for reducing error recovery time in a hard disk drive, comprising:

providing an error recovery table having a plurality of error recovery steps arranged in an error recovery step order stored in the hard disk drive;

detecting a first read error;

recovering from said first read error using said error recovery table;

determining a successful error recovery step from said error recovery step order which was successful in recovering said first read error; and

adjusting said error recovery step order in the hard disk drive based on said determining step.

2. A method, as claimed in claim 1, wherein said adjusting step includes:

moving said successful error recovery step to the beginning of said error recovery step order when said successful error recovery step is not at the beginning of said error recovery step order.

3. A method, as claimed in claim 1, wherein said adjusting step includes:

recording a first error location, a first error type, and said successful error recovery step in a first memory element; and

attempting said successful error recovery step before any other steps in said error recovery step order when a second error is detected which has said first error type and is within a predetermined distance from said first error location.

4. A method, as claimed in claim 3, wherein said predetermined distance is 100 sectors.

5. A method, as claimed in claim 3, wherein said predetermined distance is one track.

6. A method, as claimed in claim 3, wherein said predetermined distance is one disk surface.

7. A method, as claimed in claim 3, further comprising:

attempting secondly at least one of the remaining steps in said error recovery table when said successful error recovery step is not successful in recovering said second error.

8. A method, as claimed in claim 3, further comprising:

detecting a third read error;

comparing a third error type and a third error location of said third error to said first error type and first error location; and

attempting to recover said third error using said successful error recovery step when said first and third error types are similar and when said first and third error locations are within a predetermined distance from each other.

9. A method, as claimed in claim 8, wherein said predetermined distance is 100 sectors.

10. A method, as claimed in claim 8, wherein said predetermined distance is one track.

11. A method, as claimed in claim 8, wherein said predetermined distance is one disk surface.

12. A method, as claimed in claim 8, further comprising:

recovering from said third error using said error recovery table when said first and third error types are not similar or said first and third error locations are not within said predetermined distance; and

recording a third error type, a third error location, and a third error recovery step in a second memory element.

13. A method, as claimed in claim 3, wherein said recording step also includes storing an occurrence count in said first memory element.

14. A method, as claimed in claim 13, further comprising:

detecting a third read error;

comparing a third error type and a third error location of said third read error to said first error type and first error location;

recovering from said third read error using said error recovery table when said third error type is not similar to said first error type; and

recording an error type, an error location, and a recovery step for said third read error in a second memory element.

15. A method, as claimed in claim 1, wherein said adjusting step includes:

determining weight data associated with each error recovery step in said error recovery step order;

modifying said weight data based on said successful error recovery step; and

re-ordering said error recovery step order based on said modified weight data.

16. A method, as claimed in claim 15, wherein said weight data is used to determine a number of attempts each error recovery step will be given.

17. A method, as claimed in claim 15, wherein said modifying step includes:

determining a current weight for said successful error recovery step;

increasing said current weight when said current weight is less than a maximum weight; and

decreasing a weight of a last error recovery step in said error recovery table when said weight is greater than a minimum weight.

18. A method, as claimed in claim 17, wherein said re-ordering step comprises:

determining a current weight for each error recovery step in said error recovery table; and

sorting said error recovery table such that said error recovery steps are ordered according to current weight.

19. A method, as claimed in claim 17, wherein said maximum weight corresponds to eight (8) attempts for an error recovery step, and wherein said minimum weight corresponds to one (1) attempt for an error recovery step.

20. A hard disk drive, comprising:

at least one magnetic storage disk;

a transducer mounted to an actuator arm, said transducer operable to read and write data to said at least one magnetic storage disk;

control electronics operable to actuate said actuator arm and send and receive write and read signals to and from said transducer; and

a memory having a first storage area and a second storage area, said second storage area operable to store at least a first error recovery table having a first plurality of error recovery steps arranged in a first error recovery step order;

wherein said control electronics are operable to detect an error in read data and attempt to recover from said error according to said first error recovery table, and wherein said control electronics are operable to re-order said first error recovery step order based on an error recovery step used to recover from said error.

21. A hard disk drive, as claimed in claim 20, wherein said second storage area is operable to store a second error recovery table having a second plurality of error recovery steps arranged in a second error recovery step order, wherein said first error recovery table is associated with a first disk surface and said second error recovery table is associated with a second disk surface, and

wherein said control electronics are operable to determine if said error is located on said first or second disk surface and attempt to recover from said error using said first error recovery table when said error is located on said first disk surface and using said second error recovery table when said error is located on said second disk surface.

22. A hard disk drive, as claimed in claim 21, wherein said second storage area is operable to store a plurality of error recovery tables, each of said plurality of error recovery tables being associated with a data zone and having a plurality of error recovery steps arranged in respective error recovery step orders, and

wherein said control electronics are operable to determine a data zone for said error and attempt to recover from said error using the error recovery table associated with said data zone for said error.

23. A hard disk drive, as claimed in claim 22, wherein each disk surface has sixteen (16) data zones.

24. A hard disk drive, as claimed in claim 21, wherein said control electronics are operable to re-order said second error recovery step order based on an error recovery step which was successful in recovering said error.

25. A hard disk drive, as claimed in claim 20, wherein said second storage area also includes weighting data, said weighting data indicating a number of attempts each error recovery step in said error recovery table will be given.

26. A method for recovering from a read error in a hard disk drive, comprising:

detecting a first read error;

attempting a recovery of said first read error using a predetermined order of error recovery steps; and

adjusting said predetermined order of error recovery steps to create a modified order of error recovery steps when said first read error is recovered.

27. A method, as claimed in claim 26, wherein said adjusting step comprises:

determining a successful error recovery step; and

moving said successful error recovery step to the beginning of said predetermined order of error recovery steps.

28. A method, as claimed in claim 26, wherein said adjusting step comprises:

determining firstly a successful error recovery step;

determining secondly weight data associated with each of said error recovery steps;

modifying said weight data based on said successful error recovery step; and

re-ordering said order of error recovery steps based on said modified weight data.

29. A method, as claimed in claim 28, wherein said weight data is used to determine a number of attempts each error recovery step will be given.

30. A method, as claimed in claim 28, wherein said modifying step includes:

determining a current weight for said successful error recovery step;

increasing said current weight when said current weight is less than a maximum weight; and

decreasing a weight of a last error recovery step in said order of error recovery steps when said weight is greater than a minimum weight.

31. A method, as claimed in claim 30, wherein said re-ordering step comprises:

determining a current weight for each error recovery step in said order of error recovery steps; and

sorting said order of error recovery steps such that said error recovery steps are ordered according to current weight.

32. A method, as claimed in claim 31, wherein said maximum weight corresponds to eight (8) attempts for an error recovery step, and wherein said minimum weight corresponds to one (1) attempt for an error recovery step.

33. A method for recovering a read error in a hard disk drive, comprising:

detecting a read error;

determining if an error memory element contains an entry for an error having an error type and an error location corresponding to said read error;

attempting a first error recovery step associated with said error memory element when said error memory element contains an entry for an error having an error type and an error location corresponding to said read error; and

attempting an error recovery based on an error recovery table when said first error recovery step is not successful in recovering said read error, wherein said first error recovery step is omitted from said error recovery table; and

attempting an error recovery based on said error recovery table when read error does not correspond to said error type and said error location of an error memory element.

34. A method, as claimed in claim 33, further comprising:

storing an error location and an error type of said read error in an error memory element when said first error recovery step is not successful in recovering said read error.

35. A method, as claimed in claim 34, further comprising:

setting an occurrence count in said error memory element to one.

36. A method, as claimed in claim 35, further comprising:

incrementing an occurrence count in said error memory element when said first error recovery step is successful in recovering said read error.

37. A method, as claimed in claim 36, further comprising:

detecting a second read error;

recovering from said second read error using an error recovery table when a location and error type of said second error do not correspond to an error memory element; and

storing a recovery step, an error type, and an error location for said second read error in an error memory element having the lowest occurrence count.

38. A method, as claimed in claim 33, further comprising:

resetting an age data element in said error memory element.

39. A method, as claimed in claim 38, further comprising:

resetting an age data element in said error memory element when said first error recovery step is successful in recovering said read error.

40. A method, as claimed in claim 39, further comprising:

detecting a second read error;

recovering from said second read error using an error recovery table when a location and error type of said second error do not correspond to an error memory element; and

storing a recovery step, an error type, and an error location for said second read error in an error memory element having the oldest age data element.